UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/789,949	02/27/2004	Johannes Lauterbach	000005-006500US	5789

58735 7590 01/23/2009
Fountainhead Law Group P.C.
900 LAFAYETTE STREET
SUITE 509
SANTA CLARA, CA 95050

EXAMINER	
KISS, ERIC B	
ART UNIT	PAPER NUMBER
2192	
MAIL DATE	DELIVERY MODE
01/23/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/789,949	**Applicant(s)** LAUTERBACH ET AL.
	Examiner ERIC B. KISS	**Art Unit** 2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 November 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-19 and 21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-19 and 21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on November 6, 2008, has been entered. Claims 1-19 and 21 are pending.

Response to Amendment

2. The rejection of claims 12 and 17 under 35 U.S.C. § 112, second paragraph, is withdrawn in view of applicants' amendments.

Response to Arguments

3. Applicant's arguments filed November 6, 2008, have been fully considered but they are not persuasive.

As disclosed by *Sun2001* at JSP.2.1.3, an execution phase (in a runtime environment on a platform) takes place subsequent to the translation phase.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-11, 13-16, 18, and 19 are rejected under 35 U.S.C. 102(b) as being anticipated by Eduardo Pelegrí-Llopart, ed., "JavaServer Pages™ Specification," Version 1.2, August 27, 2001, Sun Microsystems, Inc. (hereinafter "Sun2001").

Regarding claim 1, *Sun2001* discloses a method for providing a library that is adapted to be instantiated into a runtime object (*see, e.g., Sun2001* at JSP.7.2 (describing tag libraries)), the method comprising:

providing a template that corresponds to the structure of the runtime object with element placeholders for elements and with attribute placeholders for attributes (*see, e.g., Sun2001* at sections JSP.1.1 and JSP.1.2 (describing templates, and adding dynamic data to template data); section JSP.2.13 (describing handling of attributes));

providing classes that form the library, wherein the classes correspond to the elements and the classes have replacement instructions for the placeholders, with the replacement instructions activated upon instantiating into the runtime object, wherein the runtime object is for subsequent execution in one of a variety of different platforms (*see, e.g., Sun2001* at Chapter JSP.7 (describing tag libraries); *Sun2001* at JSP.2.1.3 (execution subsequent to translation takes place in a runtime environment on a platform)).

Regarding claim 2, *Sun2001* further discloses the template includes element placeholders having start portions and end portions differentiated by tag types (*see, e.g., Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers and various tag types)).

Regarding claim 3, *Sun2001* further discloses the template includes element placeholders having element identification components belonging to the start portions and the end portions (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers and various tag types)).

Regarding claim 4, *Sun2001* further discloses the element placeholders include element placeholders for a root element and for a branch element, with the start portions and the end portions of the branch element placed between the start portions and the end portions of the root element (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.3 (describing tag handlers, including branches within tags (conditionals and iteration tags))).

Regarding claim 5, *Sun2001* further discloses the template includes the attribute placeholders placed between the start portions and the end portions of the element placeholders (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers accessing attributes of tags)).

Regarding claim 6, *Sun2001* further discloses the template includes code portions in the language of the runtime object placed between the start portions and the end portions of the element placeholders (*see*, *e.g.*, *Sun2001* at section JSP.7.1.3 (describing tag handlers and associated actions (code portions))).

Regarding claim 7, *Sun2001* further discloses, in providing the template, single placeholders that represent a plurality of elements include a plurality indicator for indicating that the single placeholders represent a plurality of elements (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.3.4 and JSP.7.1.3.5 (describing iterations)).

Regarding claim 8, *Sun2001* further discloses, in providing classes, the attribute placeholder changes a form of tags from tags of a first type into tags of a second type (*see*, *e.g.*, *Sun2001* at section JSP.7.1.2 (describing the translation of tags from XML to a JSP page implementation)).

Regarding claim 9, *Sun2001* further discloses providing classes comprises using XML-techniques (*see*, *e.g.*, *Sun2001* at section JSP.7.1 (a tag library is described via the tag library descriptor (TLD), and XML document)).

Regarding claim 10, *Sun2001* further discloses providing classes comprises organizing the classes in an abstract syntax tree (AST) (defined by applicant as, "any computer-internal hierarchy representation of an object, a class or a library," (Specification at p. 5, lines 4-5). (*see*, *e.g.*, *Sun2001* at section JSP.7.1.3.6 (describing a nested structure of actions to describe scoping)).

Regarding claim 11, *Sun2001* further discloses the template and classes are provided such that the library is adapted to be instantiated into a runtime object selected from the group consisting of application class file, application project file, common registry, machine specific registry, business component, and website layout (*see*, *e.g.*, *Sun2001* at section JSP.1.1 (describing general uses of JSP, including web applications (business component and website layout))).

Regarding claim 13, *Sun2001* further discloses each element has associated attributes (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers accessing attributes of tags)), further comprising:

identifying data for the attributes associated with each of the elements (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers accessing attributes of tags)); and instantiating the classes by activating the replacement instructions to replace the attribute placeholders with the data (*see, e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers accessing attributes of tags)).

Regarding claim 14, Sun2001 discloses an article of manufacture comprising a computer-usable medium storing computer-readable program code for causing a processor to perform operations (*see, e.g.*, *Sun2001* at section JSP.2.1.3 (describing generally execution involving a server and client, necessarily requiring processors and memory)) comprising:

providing a runtime object having elements and attributes, with each element having associated ones of the attributes (*see, e.g.*, *Sun2001* at sections JSP.2.1.3 (describing translation and execution of tags) and JSP.7.1.3 (describing tag handlers accessing attributes of tags));

pre-assembling the runtime object using classes in a library, wherein the classes correspond to the elements, the classes include replacement instructions for attribute placeholders, and the classes are based on a template that corresponds to a structure of the runtime object, with the template including element placeholders for the elements and attribute placeholders for the attributes (*see, e.g.*, *Sun2001* at sections JSP.1.1 and JSP.1.2 (describing templates, and adding dynamic data to template data); section JSP.2.13 (describing handling of attributes); Chapter JSP.7 (describing tag libraries));

identifying data for the attributes associated with each of the elements (*see, e.g.*, *Sun2001* at section (describing tag handlers accessing attributes of tags)); and

instantiating the classes by activating the replacement instructions to replace the attribute placeholders with the data, wherein instantiating the classes results in a runtime object for subsequent execution in one of a variety of different platforms (*see*, *e.g.*, *Sun2001* at sections JSP.2.1.3 (describing translation and execution of tags) and JSP.7.1.3 (describing tag handlers accessing attributes of tags); *Sun2001* at JSP.2.1.3 (execution subsequent to translation takes place in a runtime environment on a platform)).

Regarding claim 15, *Sun2001* further discloses the template includes element placeholders having start portions and end portions and the attribute placeholders are placed between the start portions and the end portions of the element placeholders (*see*, *e.g.*, *Sun2001* at sections JSP.7.1.2.1 and JSP.7.1.3 (describing tag handlers accessing attributes of tags)).

Regarding claim 16, *Sun2001* further discloses the template includes code portions in the language of the runtime object placed between the start portions and the end portions of the element placeholders (*see*, *e.g.*, *Sun2001* at section JSP.7.1.3 (describing tag handlers and associated actions (code portions))).

Regarding claim 18, *Sun2001* discloses a computer program stored on a computer-readable medium and comprising processor instructions for providing a library adapted to be instantiated into a runtime object (*see*, *e.g.*, *Sun2001* at sections JSP.2.1.3 (describing generally execution involving a server and client, necessarily requiring processors and memory); *Sun2001* at section JSP.7.2 (describing tag libraries)), the processor instructions comprising:

first instructions for providing a template that corresponds to a structure of the runtime object with element placeholders for elements and with attribute placeholders for attributes (*see*, *e.g.*, *Sun2001* at sections JSP.1.1 and JSP.1.2 (describing templates, and adding dynamic data to template data); section JSP.2.13 (describing handling of attributes)); and

second instructions for providing classes that form the library, wherein the classes correspond to the elements and the classes have replacement instructions for the placeholders that are activated upon instantiating into the runtime object, wherein instantiating the classes results in a runtime object for subsequent execution in one of a variety of different platforms (*see*, *e.g.*, *Sun2001* at Chapter JSP.7 (describing tag libraries); *Sun2001* at JSP.2.1.3 (execution subsequent to translation takes place in a runtime environment on a platform)).

Regarding claim 19, *Sun2001* further discloses the library is adapted to be instantiated into a runtime object selected from the group consisting of application class file, application project file, common registry, machine specific registry, business component, and website layout (describing general uses of JSP, including web applications (business component and website layout))).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 12, 17, and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Sun2001*, as applied above to claims 1, 14, and 18, in view of "C++ Server Pages," Micronovae Ltd, 12/3/2002 [online], accessed 1/15/2009, Retrieved from Internet <URL:

http://web.archive.org/web/20021203004530/http://www.micronovae.com/CSP.html), 3 pages (hereinafter "*CSP2002*").

Regarding claims 12, 17, and 21, although *Sun2001* discloses the use JAVASERVER PAGES server-side scripting technology with a JAVA programming language runtime, *CSP2002* teaches that it has been known to apply such server-side scripting techniques to a C++ runtime environment (*e.g.*, *CSP2002* at p.1), and therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made for the library to be adapted to be instantiated into a C++ or C runtime object to gain advantages taught, for example, by *CSP2002* at p. 1.

Conclusion

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric B. Kiss whose telephone number is (571) 272-3699. The examiner can normally be reached on Tue. - Fri., 7:00 am - 4:30 pm. The examiner can also be reached on alternate Mondays.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam, can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Eric B. Kiss/
Eric B. Kiss
Primary Examiner, Art Unit 2192